12

Stanford Artificial Intelligence Laboratory
Memo AIM-299

Computer Science Department
Report No. STAN-CS-77-614

May 1977

THE CONVERGENCE OF FUNCTIONS TO FIXEDPOINTS OF RECURSIVE DEFINITIONS

by

ZOHAR MANNA and ADI SHAMIR

See back page for 1473

Research sponsored by

Advanced Research Projects Agency

COMPUTER SCIENCE DEPARTMENT
Stanford University

Stanford Artificial Intelligence Laboratory
Memo AIM-299

May 1977

Computer Science Department
Report No. STAN-CS-77-614

THE CONVERGENCE OF FUNCTIONS TO FIXEDPOINTS OF RECURSIVE DEFINITIONS

by

ZOHAR MANNA
Artificial Intelligence Lab
Stanford University
Stanford, Cal. 94305

ADI SHAMIR
Department of Mathematics
M.I.T.
Cambridge, Mass. 02139

Abstract: The classical method for constructing the least fixedpoint of a recursive definition is to generate a sequence of functions whose initial element is the totally undefined function and which converges to the desired least fixedpoint. This method, due to Kleene, cannot be generalized to allow the construction of other fixedpoints.

In this paper we present an alternate definition of convergence and a new *fixedpoint access* method of generating sequences of functions for a given recursive definition. The initial function of the sequence can be an arbitrary function, and the sequence will always converge to a fixedpoint that is "close" to the initial function. This defines a monotonic mapping from the set of partial functions onto the set of all fixedpoints of the given recursive definition.

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract MDA 903-76-C-0206. The authors were also affiliated with the Department of Applied Mathematics of the Weizmann Institute of Science during the period of this research.

Manna & Shamir

Contents:

Introduction

A recursive definition of the form $F(x) \equiv \tau[F](x)$ (where F is a function variable and τ is a functional) can be considered as an implicit functional equation. In general, such a functional equation may have many possible solutions (*fixedpoints*), all of which satisfy the relations dictated by the recursive definition. Of all these fixedpoints, only one, the ***least fixedpoint***, has been studied thoroughly; however, recursive definitions have other interesting solutions (e.g., the ***optimal fixedpoint*** discussed in Manna and Shamir [1976]). By considering the properties of the entire set of fixedpoints, a unified theory for the various fixedpoint approaches can be developed.

One of the most fundamental results in the theory of recursive definitions is *Kleene's Theorem* which states that (under suitable conditions) the least fixedpoint is the least upper bound (*lub*) of the sequence Ω, $\tau[\Omega]$, $\tau^2[\Omega]$, ... , where the initial function Ω is the totally undefined function. This theorem gives a constructive method by which the least fixedpoint can be "accessed" from the initial function Ω.

The purpose of this paper is to generalize Kleene's Theorem so that arbitrary fixedpoints of a recursive definition can be accessed. This is done by altering Kleene's access method in three ways: by allowing an arbitrary initial function, by generating the corresponding sequence of functions in a different manner, and by introducing a modified notion of convergence.

Part I contains all the preliminary definitions and results. Our, slightly nonstandard, model of recursive definitions is presented in Section 1. In Section 2 we prove some properties of functionals in this model, and in Section 3 we study the elementary closure properties of three important sets of functions: *fixedpoints*, *prefixedpoints*, and *postfixedpoints*.

Our generalization of Kleene's Theorem is discussed in Part II. In Section 4, we consider the behavior of Kleene's "direct" access method for initial functions other than Ω . In particular, we show that this generalized sequence of functions may fail to converge, but whenever it converges the limit is a fixedpoint which is "close" to the initial function.

More general types of access methods are defined in Section 5. In essence, each such method defines a sequence of transformations which should be applied to the initial function. These transformations are defined in terms of the three basic operations: ***functional application***, ***glb***, and ***lub***. Among the access methods, we pay special attention to the "descending" access method. The sequences of functions generated by this method always converge, but their limit need not be a fixedpoint.

Finally, in Section 6, we show that under the composition of the "descending" and "direct"

access methods, *any* initial function converges to a "close" fixedpoint. We then prove that no single access method can enjoy this property, and thus the composition of methods is essential.

Part I: Recursive Definitions and Their Fixedpoints

1. The Model

1.1 The Basic Domains

The purpose of this subsection is to introduce the basic terminology about partially ordered sets used throughout this paper.

Definition: A binary relation $\sqsubseteq$ over a nonempty set S is a *partial ordering* of S if $\sqsubseteq$ is a reflexive, transitive and antisymmetric relation. The pair $(S,\sqsubseteq)$ is called a *partially ordered set* (*poset*).

Definition: Let $(S,\sqsubseteq)$ be a poset. For a subset A of S , an element $x \in S$ is called:

(a) *least* if $x \in A$ and for all $y \in A$, $x \sqsubseteq y$;

(b) *greatest* if $x \in A$ and for all $y \in A$, $y \sqsubseteq x$;

(c) *minimal* if $x \in A$ and there is no $y \in A$, $y \neq x$ for which $y \sqsubseteq x$;

(d) *maximal* if $x \in A$ and there is no $y \in A$, $y \neq x$ for which $x \sqsubseteq y$;

(e) *lower bound* if for all $y \in A$, $x \sqsubseteq y$;

(f) *upper bound* if for all $y \in A$, $y \sqsubseteq x$;

(g) *greatest lower bound* (*glb*) if x is a lower bound of A , and for any other lower bound y of A , $y \sqsubseteq x$;

(h) *least upper bound* (*lub*) if x is an upper bound of A , and for any other upper bound y of A , $x \sqsubseteq y$.

Definition: A *semilattice* is a poset $(S,\sqsubseteq)$ in which any two elements in S have a *glb*. A *complete semilattice* is a poset $(S,\sqsubseteq)$ in which any nonempty subset of S has a *glb*.

Such structures are usually called "lower semilattice" and "complete lower semilattice". The

notions of "upper semilattice" and "complete upper semilattice" are similarly defined with the *glb* replaced by *lub* in the definition. However, we omit the word "lower" since in this paper we work exclusively with lower semilattices and no confusion is caused.

Definition: A subset A of S in a semilattice $(S, \sqsubseteq)$ is said to be *consistent* if it has an *lub*. An element $x \in S$ is said to be *consistent with* an element $y \in S$ if the set $\{x,y\}$ is consistent.

Semilattices may contain both consistent and inconsistent sets. The binary relation of being "consistent with" is clearly reflexive and symmetric, but not necessarily transitive. Note that if the semilattice is complete, the existence of some upper bound implies the existence of a *lub*. Any subset of a consistent set is also consistent in this case, but pairwise consistency of elements does not imply the consistency of the set as a whole.

Definition: A sequence $x_0, x_1, x_2, \ldots$ of elements in a poset S is an *ascending* (*descending*) *chain* if $x_i \sqsubseteq x_{i+1}$ ($x_{i+1} \sqsubseteq x_i$) for all i. The sequence is a *chain* if it is either an ascending or a descending chain.

Definition: A *flat semilattice* is a semilattice in which all chains contain at most two distinct elements.

It is clear that any flat semilattice is complete; it contains a *bottom element* ω (which satisfies $\omega \sqsubseteq d$ for all d), and all the other elements are unrelated. The importance of this structure in the theory of computation stems from the fact that they represent the two-state discrete type of knowledge which often occurs during a computation: A variable either contains a well-characterized value or has an undefined value (if used without proper initialization); an operation (such as a division of two numbers) may either yield a definite result or terminate as "illegal"; a procedure call may either return a proper result or loop forever. In all these cases, one possible extreme is a totally defined entity, while absolutely nothing is known about the other (besides its very "undefinedness").

All the basic domains considered in this paper are flat semilattices, denoted by D. Two domains of special importance are the *Boolean domain* $B = (\{\omega, true, false\}, \sqsubseteq)$ and the domain of natural numbers $N = (\{\omega, 0, 1, 2, \ldots\}, \sqsubseteq)$.

1.2 Higher Type Objects

In this section we inductively define the objects of all finite types over the basic domain D_i. The two basic notions used, that of a convergent sequence and that of a continuity, are defined in a nonstandard way. The classical definition of these notions is heavily oriented towards the needs of the least fixedpoint approach; we need more balanced definitions in order to construct a general fixedpoint theory of recursive definitions. In particular, we no longer concentrate on ascending chains and their *lub*, but consider also descending chains and their *glb*, as well as more general forms of convergence.

Definition: A mapping $\phi : A \to B$ between posets is *monotonic* if $\phi(x) \sqsubseteq \phi(y)$ in B whenever $x \sqsubseteq y$ in A.

Definition: The set of (finite) types is defined inductively as follows:

(*i*) Any basic domain D_i is a type; the objects of this type are the elements of D_i.

(*ii*) If $\sigma_1, ..., \sigma_k$ are types, so is $\sigma_1 \times ... \times \sigma_k$; the objects of this type are the vectors $(x_1, ..., x_k)$ where each x_i is an object of type σ_i.

(*iii*) If σ_1, σ_2 are types, so is $[\sigma_1 \to \sigma_2]$; the objects of this type are the monotonic mappings from objects of type σ_1 to objects of type σ_2.

There is a natural way to extend the $\sqsubseteq$ relation to the set of objects of any finite type, using the following inductive definition:

Definition:

(*i*) If $\bar{x} \equiv (x_1, ..., x_k)$ and $\bar{y} \equiv (y_1, ..., y_k)$ are objects of type $\sigma_1 \times ... \times \sigma_k$, then $\bar{x} \sqsubseteq \bar{y}$ iff for all $1 \le i \le k$, $x_i \sqsubseteq y_i$ as objects of type σ_i.

(*ii*) If x and y are objects of type $[\sigma_1 \to \sigma_2]$, then $x \sqsubseteq y$ iff for any fixed object z of type σ_1, $x(z) \sqsubseteq y(z)$ as objects of type σ_2.

It is easy to see that the set of objects of any finite type is a complete semilattice under this relation.

The notions of a convergent sequence and limit are usually identified with those of an ascending chain and *lub*, respectively. Our definition of these notions is more inclusive:

Definition: A sequence of objects $\{x_j\}$ of some finite type σ is said to *converge* to the object x_∞ of type σ, written as $x_\infty \equiv lim\{x_j\}$, if:

(*i*) σ is some basic domain D_i, and all the elements in $\{x_j\}$ are equal to x_∞ from some index j_0 onwards.

(*ii*) σ is $\sigma_1 \times \ldots \times \sigma_k$ and for any $1 \le i \le k$, $x_\infty{}^i \equiv lim\{x_j{}^i\}$ (where $x_j{}^i$ is the i-th component of x_j).

(*iii*) σ is $[\sigma_1 \to \sigma_2]$ and for any fixed object z of type σ_1, $x_\infty(z) \equiv lim\{x_j(z)\}$ (these are objects of type σ_2, for which the notion of convergence is already defined).

Parts (*ii*) and (*iii*) in this definition are standard, and once we define our notion of convergence in the basic domains, it is carried over to all finite types. It is easy to see that any ascending or descending chain of any type is a convergent sequence (with *lub* or *glb*, respectively, as limits). The following example shows that the converse is not true:

Example 1: Let $\{f_i\}$ be a sequence of objects of type $[N \to N]$, defined by:

$$f_i(x) \equiv \begin{cases} i & \text{if } x \ge i \\ 0 & \text{if } x < i \\ \omega & \text{if } x \equiv \omega \end{cases}$$

No two elements in the sequence $\{f_i\}$ are related by $\sqsubseteq$, but the sequence converges to the object *zero* of type $[N \to N]$

$$zero(x) = \begin{cases} \omega & \text{if } x \equiv \omega \\ 0 & \text{otherwise} \end{cases}$$

This follows immediately from the fact that for any argument x of type N, the sequence $\{f_i(x)\}$ of elements of type N is convergent, i.e., its elements are 0 for all sufficiently high i. □

Using the notion of a convergent sequence, we can define our notion of continuity:

Definition:

(*i*) An object $(x_1, ..., x_k)$ of type $\sigma_1 \times ... \times \sigma_k$ is continuous if all the objects x_i are continuous.

(*ii*) An object x of type $[\sigma_1 \to \sigma_2]$ is continuous if for any convergent sequence $\{z_j\}$ of objects of type σ_1, the sequence $\{x(z_j)\}$ of objects of type σ_2 is convergent and $x(lim\{z_j\}) \equiv lim\{x(z_j)\}$.

Since the notion of a convergent sequence is more inclusive than that of a chain, our notion of continuous objects (i.e., of limit-preserving mappings) is potentially more restrictive than the standard notion of chain-continuity. The following example shows that in fact an object can preserve the *lub* and *glb* of ascending and descending chains, and still be noncontinuous in our system:

Example 2: Let f be an object of type $[N \to N]$. We say that f is *closed* if the sequence $\{x_i\}$ defined by

$$x_0 \equiv 0 \text{ and } x_{i+1} \equiv f(x_i) \text{ (i.e., } x_i \equiv f^{(i)}(0) \text{)}$$

consists of a finite number of distinct elements, none of which is ω. It is clear that a necessary and sufficient condition for a function f to be closed is the existence of numbers $0 \le i < j$ such that $f^{(i)}(0) \equiv f^{(j)}(0) \not\equiv \omega$, in which case the sequence $\{x_i\}$ is periodic from some point onwards.

Let the object Θ of type $[[N \to N] \to B]$ be defined as follows:

$$\theta[f] \equiv \begin{cases} \mathit{true} & \text{if } f \text{ contains a finite sequence of pointers} \\ \omega & \text{otherwise} \end{cases}$$

The object Θ preserves the *lub* and *glb* of ascending and descending chains, since the finite number of values $f(x_i)$ which constitute a sequence of pointers are either constructed or destroyed at some finite point in any chain $\{f_i\}$, and thus $\Theta[lim\{f_i\}] \equiv \Theta[f_k]$ for some k.

However, Θ is not continuous in our model. Consider, for example, the following sequence of objects $\{f_i\}$:

$$f_i(x) \equiv \begin{cases} x + 1 & \text{if } x < i \\ x & \text{if } x \ge i \end{cases}$$

The sequence converges to the object

$$f_\infty(x) \equiv x + 1.$$

It is easy to see that $\Theta[f_\infty]$ is ω, while for any i, $\Theta[f_i]$ is *true*. Thus $\Theta[lim\{f_i\}] \not\equiv lim\{\Theta[f_i]\}$ and Θ is not continuous. □

From now on, we shall be interested mainly in the lower three types of objects: values (objects of type D_i), functions (objects of type $[D_1 \times ... \times D_k \to D_0]$, and (single-argument) functionals (objects of type $[[D^1_1 \times ... \times D^1_k \to D^1_0\,] \to [D^2_1 \times ... \times D^2_l \to D^2_0\,]]$). Since we shall not deal with systems of recursive definitions, we do not have to consider multi-argument functionals (for which the fixedpoint theory obtained is somewhat different).

1.3 Term Functionals and Recursive Definitions

Among all the functionals τ, we shall be interested mainly in *term functionals*, which are syntactically expressed as compositions of constants, monotonic base functions g_i, a function variable F, and individual variables x_i. Associated with each symbol (including the variables) is a type, and the composition of these types must be legal.

Example 3: A term of the form

$$\textbf{if } g(x_1\,, x_1) \textbf{ then } x_2 \textbf{ else } g(x_2\,, x_3)$$

can be legal only if the types of x_1, x_2, and x_3 are the boolean semilattice B, and the type of g is $[B \times B \to B]$. This can be shown by the following argument:

Since $g(x_1,x_1)$ appears in the **if** part, the range of this term must be B. Since the two subterms x_2 and $g(x_2,x_3)$ must have identical ranges, the type of x_2 is necessarily B. Therefore the type of g is of the form $[B \times ? \to B]$. In order to make the term $g(x_1\,, x_1)$ legal, x_1 must be of type B, implying that "?" is also B. We can thus conclude (from the term $g(x_2,x_3)$) that x_3 is also of type B. □

A term functional is denoted by $\tau[F](x_1, ..., x_k)$, where $x_1, ..., x_k$ are all the individual variables occurring in it, in some order. It can be interpreted as a functional in the following way:

Given a function f and an argument vector $\bar{d} = (d_1, ..., d_k)$ (of the appropriate types), the value of $\tau[f](\bar{d})$ is the object obtained by evaluating the variable-free term in which F is interpreted as f and x_i is interpreted as d_i. The function $\tau[f]$ to which f is mapped under τ is the function abstraction $\lambda\bar{x}\ \tau[f](\bar{x})$. The fact that τ maps monotonic functions to monotonic functions is immediate from the fact that all the base functions in τ are monotonic, and the set of monotonic functions is closed under composition.

Definition: A *recursive definition* is an equation of the form

$$F(\bar{x}) \equiv \tau[F](\bar{x}),$$

where τ is a term functional.

In order to make this equation meaningful, τ must map functions of the appropriate type $[D_1 \times ... \times D_k \to D_0]$ to functions of the same type.

2 Properties of Term Functionals

The fact that term functionals are monotonic mappings which preserve the *lub* of ascending chains is one of the oldest and most basic results in the recursive definitions theory. In a simple form it appears in Kleene [1952], while a detailed proof of this result for a model of functionals which is quite similar to ours appears in Cadiou [1972]. In this section we prove the stronger result of continuity in our model, and discuss the behavior of term functionals under the *glb* and *lub* operations over arbitrary sets of functions (rather than over chains).

2.1 The Continuity of Term Functionals

Under the classical definition of continuity, any mapping which preserves the *lub* of ascending chains is necessarily monotonic. However, a mapping Θ can preserve the limits of convergent sequences without preserving a *lub* of chains, or without being monotonic at all. This happens, for example, when Θ maps an ascending chain $\{x_i\}$ into a descending chain $\{\Theta(x_i)\}$ provided that

$$\Theta(lim\{x_i\}) \equiv \Theta(lub\{x_i\}) \equiv glb\{\Theta(x_i)\} \equiv lim\{\Theta(x_i)\}.$$

The property of continuity is thus totally independent from the property of monotonicity in our model.

We now prove the basic result:

Theorem 1: Let τ be a term functional and $\{f_i\}$ a convergent sequence. Then $\{\tau[f_i]\}$ is a convergent sequence and

$$lim\{\tau[f_i]\} \equiv \tau[lim\{f_i\}]\,.$$

Proof: The proof is by induction on the structure of τ, using the fact that term functionals contain finitely many basic constructs. Note that the monotonicity of these constructs is not used at all.

If τ is a variable x_i or constant c, the proof is trivial.

If τ is of the form $g(\tau_1, ..., \tau_n)$, we may apply the induction hypothesis that all the subterms τ_i are continuous. Let $\bar{x}$ be fixed. Then for any $1 \le k \le n$, there is an index j_k such that

$$\tau_k[f_j](\bar{x}) \equiv \tau_k[lim\{f_i\}](\bar{x}) \quad \text{for all } j \geq j_k .$$

Let j_0 be $max(j_1, ..., j_n)$. Then for all $j \geq j_0$:

$$\begin{aligned}
\tau[f_j](\bar{x}) &\equiv g(\tau_1[f_j](\bar{x}), ..., \tau_n[f_j](\bar{x})) \\
&\equiv g(\tau_1[lim\{f_i\}](\bar{x}), ..., \tau_n[lim\{f_i\}](\bar{x})) \\
&\equiv \tau[lim\{f_i\}](\bar{x}).
\end{aligned}$$

Finally, if τ is of the form $F(\tau_1, ..., \tau_n)$, we define j_0 in exactly the same way as before. We denote the vector ($\tau_1[lim\{f_i\}](\bar{x})$, ... , $\tau_n[lim\{f_i\}](\bar{x})$) by $\bar{y}$, and thus by the definition of τ,

$$\tau[lim\{f_i\}](\bar{x}) \equiv (lim\{f_i\})(\bar{y}) .$$

Since $\{f_i\}$ is a convergent sequence, there is some j_0' such that

$$f_j(\bar{y}) \equiv (lim\{f_i\})(\bar{y}) \quad \text{for all } j \geq j_0' .$$

Let j_0'' be $max(j_0, j_0')$. Then we have, for all $j \geq j_0''$:

$$\begin{aligned}
\tau[f_j](\bar{x}) &\equiv f_j(\tau_1[f_j](\bar{x}), ..., \tau_n[f_j](\bar{x})) \\
&\equiv f_j(\tau_1[lim\{f_i\}](\bar{x}), ..., \tau_n[lim\{f_i\}](\bar{x})) \\
&\equiv f_j(\bar{y}) \equiv (lim\{f_i\})(\bar{y}) \equiv \tau[lim\{f_i\}](\bar{x}).
\end{aligned}$$

Q.E.D.

Some of the consequences of Theorem 1 are:

Corollary: Let τ be a term functional. Then:

(i) If $\{f_i\}$ is an ascending chain, then $\{\tau[f_i]\}$ is an ascending chain and $lub\{\tau[f_i]\} \equiv \tau[lub\{f_i\}]$.

(ii) If $\{f_i\}$ is a descending chain, then $\{\tau[f_i]\}$ is a descending chain and $glb\{\tau[f_i]\} \equiv \tau[glb\{f_i\}]$.

Proof:

(*i*) Any ascending chain $\{f_i\}$ is a convergent sequence, and $lub\{f_i\} \equiv lim\{f_i\}$. Since term functionals are monotonic, $\{\tau[f_i]\}$ is also an ascending chain and $lub\{\tau[f_i]\} \equiv lim\{\tau[f_i]\}$. By Theorem 1,

$$lub\{\tau[f_i]\} \equiv lim\{\tau[f_i]\} \equiv \tau[lim\{f_i\}] \equiv \tau[lub\{f_i\}].$$

(*ii*) The proof is similar.

Q.E.D.

2.2 Behavior Under the *glb* and *lub* Operations

Lemma 1: For any monotonic functional τ:

(*i*) If $\{f_\alpha\}$ is a nonempty set of functions, then

$$\tau[glb\{f_\alpha\}] \sqsubseteq glb\{\tau[f_\alpha]\}.$$

(*ii*) If $\{f_\alpha\}$ is a consistent set of functions, then so is $\{\tau[f_\alpha]\}$, and

$$lub\{\tau[f_\alpha]\} \sqsubseteq \tau[lub\{f_\alpha\}].$$

Proof:

(*i*) Since τ is monotonic and $glb\{f_\alpha\} \sqsubseteq f_\alpha$ for all α, $\tau[glb\{f_\alpha\}] \sqsubseteq \tau[f_\alpha]$ for all α. Thus $\tau[glb\{f_\alpha\}]$ is a lower bound of the set $\{\tau[f_\alpha]\}$, and therefore $\tau[glb\{f_\alpha\}] \sqsubseteq glb\{\tau[f_\alpha]\}$.

(*ii*) Since $\{f_\alpha\}$ is consistent, its *lub* exists. By the same procedure as above, $\tau[lub\{f_\alpha\}]$ can be shown to be an upper bound of $\{\tau[f_\alpha]\}$. In our model this implies the existence of $lub\{\tau[f_\alpha]\}$, and we have $lub\{\tau[f_\alpha]\} \sqsubseteq \tau[lub\{f_\alpha\}]$. Q.E.D.

According to corollary (*ii*) of Theorem 1, the inequality $\tau[glb\{f_\alpha\}] \sqsubseteq glb\{\tau[f_\alpha]\}$ becomes an equality if τ is a term functional and $\{f_\alpha\}$ is a descending chain. This result can be strengthened by showing that for a wide subclass of term functionals in our model, the words "a descending chain" can be replaced by "a consistent set". Mappings which preserve the *glb* of

consistent sets of arguments are defined and studied in Berry [1976] in connection with the bottom-up computations of least fixedpoints.

The dual property of preserving the *lub* of arbitrary consistent sets of functions holds only for a very restricted subclass of term functionals (mainly those in which the term $\tau[F](x)$ can be simplified, for any given x_0, to a term with a single occurrence of F). The problem in more realistic cases is demonstrated by the following example:

Example 4: Let τ be the following functional over the natural numbers:

$$\tau[F](x): \quad F(x+1) \cdot F(x+2)$$

(where $0 \cdot \omega \equiv \omega \cdot 0 \equiv \omega$). Define the functions

$$f_1(x) \equiv \begin{cases} 0 & \text{if } x \text{ is even} \\ \omega & \text{otherwise} \end{cases} \qquad f_2(x) \equiv \begin{cases} 0 & \text{if } x \text{ is odd} \\ \omega & \text{otherwise} \end{cases}$$

Then f_1 and f_2 are consistent, but

$$lub\{\tau[f_1], \tau[f_2]\} \equiv lub\{\Omega, \Omega\} \equiv \Omega \not\equiv zero \equiv \tau[zero] \equiv \tau[lub\{f_1, f_2\}].$$

$\square$

3. Properties of Fixedpoints, Prefixedpoints and Postfixedpoints

A recursive definition $F(x) \equiv \tau[f](x)$ can be considered as an implicit functional equation in F. With each such recursive definition, we associate three important sets of functions: fixedpoints, prefixedpoints, and postfixedpoints.

3.1 Closure Properties

Definition:

(*i*) A partial function f is a *fixedpoint* of a functional τ, or of a recursive definition $F(\bar{x}) \equiv \tau[F](\bar{x})$, if $f \equiv \tau[f]$. The set of all fixedpoints of τ is denoted by $FXP(\tau)$.

(*ii*) A partial function f is a *prefixedpoint* of a functional τ, or of a recursive definition $F(\bar{x}) \equiv \tau[F](\bar{x})$, if $f \sqsubseteq \tau[f]$. The set of all prefixedpoints of τ is denoted by $PRE(\tau)$.

(*iii*) A partial function f is a *postfixedpoint* of a functional τ, or of a recursive definition $F(\bar{x}) \equiv \tau[F](\bar{x})$, if $\tau[f] \sqsubseteq f$. The set of all postfixedpoints of τ is denoted by $POST(\tau)$.

Example 5: Consider the following recursive definition, in which F is of type $[N \times N \to N]$:

$$F(x,y) \equiv \textbf{if } x = 0 \textbf{ then } y \textbf{ else } F(F(x,y-1),F(x-1,y)).$$

The following three (quite different) functions are all fixedpoints of this recursive definition, as can be shown by direct substitution:

(*i*) $f_1(x,y) \equiv \text{if } x = 0 \text{ then } y \text{ else } \omega$;

(*ii*) $f_2(x,y) \equiv \text{if } x \geq 0 \text{ then } y \text{ else } \omega$;

(*iii*) $f_3(x,y) \equiv max(x,y)$.

The recursive definition has infinitely many more fixedpoints. A whole family of such fixedpoints is

(*iv*) $f_a(x,y) \equiv \textbf{if } x = 0 \textbf{ then } y \textbf{ else } a(x)$

where $a(x)$ is any function over the natural numbers satisfying

$$a(x) \neq 0 \text{ and } a(a(x)) = a(x) \text{ for all } x > 0.$$

Examples of functions satisfying this conditions are the identity function, any nonzero constant function, or the function which assigns to any $n \geq 2$ its greatest prime factor (with $a(1) = 1$).

The totally undefined function Ω is clearly a prefixedpoint of any recursive definition; in our case it is an example of a prefixedpoint which is not a fixedpoint.

An infinite class of postfixedpoints which are not fixedpoints of this recursive definition is

$$g_i(x, y) = \begin{cases} y & \text{if } 0 \leq x \leq i \\ \omega & \text{otherwise} \end{cases}$$

for all $i \geq 1$. □

By definition, it is clear that a partial function f is a fixedpoint of a functional τ if and only if it is both a prefixedpoint and a postfixedpoint of τ (that is, $FXP(\tau) = PRE(\tau) \cap POST(\tau)$).

In this section we summarize the closure properties of the sets $FXP(\tau)$, $PRE(\tau)$ and $POST(\tau)$ under the operations *lub*, *glb* and *lim*. These properties belong to the "folklore" of known but seldom stated facts about recursive definitions.

Lemma 2: For any monotonic functional τ:

(*i*) τ maps $FXP(\tau)$, $PRE(\tau)$ and $POST(\tau)$ into themselves.

(*ii*) $PRE(\tau)$ is closed under the *lub* operation over consistent sets.

(*iii*) $POST(\tau)$ is closed under the *glb* operation over nonempty sets.

Proof:

(*i*) Immediate from the monotonicity of τ.

(*ii*) Let $\{f_\alpha\}$ be a consistent subset of $PRE(\tau)$. Then for each α, $f_\alpha \sqsubseteq \tau[f_\alpha]$. Since $lub\{f_\alpha\}$ exists, $f_\alpha \sqsubseteq lub\{f_\alpha\}$, and τ is monotonic, we have

$$f_\alpha \sqsubseteq \tau[f_\alpha] \sqsubseteq \tau[lub\{f_\alpha\}].$$

Thus $\tau[lub\{f_\alpha\}]$ is an upper bound of $\{f_\alpha\}$, and therefore

$$lub\{f_\alpha\} \sqsubseteq \tau[lub\{f_\alpha\}].$$

In other words, $lub\{f_\alpha\}$ is also a prefixedpoint.

(*iii*) Similar. Q.E.D

It is not hard to show by appropriate counterexamples that PRE(τ) need not be closed under *glb*, POST(τ) need not be closed under *lub*, and FXP(τ) need not be closed under either operation.

Let us turn now to consider yet another operation -- the *lim* of convergent sequences.

Lemma 3: For any term functional τ, FXP(τ), PRE(τ) and POST(τ) are all closed under the *lim* operation.

Proof:

(*i*) Let $\{f_i\}$ be a convergent sequence of fixedpoints of τ. By Theorem 1 we have:

$$\tau[lim\{f_i\}] \equiv lim\{\tau[f_i]\} \equiv lim\{f_i\},$$

and thus $lim\{f_i\}$ is also a fixedpoint of τ.

(*ii*) Let $\{f_i\}$ be a convergent sequence of prefixedpoints of τ. Then for any i, $f_i \sqsubseteq \tau[f_i]$. By the definition of the *lim* operation we have

$$lim\{f_i\} \sqsubseteq lim\{\tau[f_i]\},$$

By Theorem 1, $lim\{\tau[f_i]\}$ exists and $lim\{\tau[f_i]\} \equiv \tau[lim\{f_i\}]$. Thus

$$lim\{f_i\} \sqsubseteq \tau[lim\{f_i\}],$$

or equivalently $lim\{f_i\}$ is a prefixedpoint of τ.

(*iii*) Similar to (*ii*). Q.E.D.

An important special case is:

Corollary: For a term functional τ, FXP(τ), PRE(τ) and POST(τ) are all closed under the *lub* and *glb* of ascending and descending chains.

3.2 Maximal and Minimal Fixedpoints

We turn now to study those fixedpoints located at the extreme ends of FXP(τ) -- the maximal and the minimal fixedpoints of τ.

As usual, a *maximal fixedpoint* of τ is defined to be a fixedpoint which is not less defined than any other fixedpoint of τ. The set of all maximal fixedpoints is denoted by MAX(τ).

A basic property of MAX(τ) is:

Theorem 2: For a monotonic functional τ,
if $f \in$ PRE(τ) then $f \sqsubseteq g$ for some $g \in$ MAX(τ).

Proof: This is quite a straightforward application of Zorn's Lemma which states that if (S,<) is a nonempty partially ordered set in which any totally ordered subset has an upper bound, then S contains a maximal element (see e.g. Dugundji [1966]).

For our purposes, we take the set

$$S = \{ h \in \mathrm{PRE}(\tau) \mid f \sqsubseteq h \}$$

with the standard partial ordering $\sqsubseteq$. This set is not empty since $f \in S$. If S_1 is a totally ordered subset of S, it is in particular consistent, and thus *lub*S_1 exists. By Lemma 2(*ii*) *lub*S_1 is a prefixedpoint of τ, and it clearly satisfies $f \sqsubseteq$ *lub*S_1. Thus *lub*$S_1 \in S$ and therefore the subset S_1 has an upper bound in S.

We may now apply Zorn's Lemma, which guarantees the existence of a maximal element $g \in S$. By definition, $f \sqsubseteq g$ and $g \sqsubseteq \tau[g]$. To show that g is a fixedpoint of τ, we note that by Lemma 2(*i*), $\tau[g]$ is also a prefixedpoint of τ in S, and thus the assumption that $g \neq \tau[g]$ contradicts the maximality of g in S. Q.E.D.

Since for any functional τ, PRE(τ) is nonempty ($\Omega \in$ PRE(τ)), we have:

Corollary: For any monotonic functional τ, MAX(τ) is not empty.

This corollary guarantees the existence of at least one maximal fixedpoint, but it need not be

unique. As a matter of fact, monotonic functionals may have any number of maximal fixedpoints in our semilattice model.

Let us consider now the minimal fixedpoints of a monotonic functional τ. The main result (the *Least Fixedpoint Theorem*) states that a monotonic functional τ has a least (and thus a unique minimal) fixedpoint, which we denote by $lfxp(\tau)$. This is a classical theorem, and it has two well-known types of proofs:

(*i*) (A nonconstructive proof, due to Tarski [1955]): In a model in which τ is defined over a complete lattice (rather than a complete semilattice) of elements, one can take the *glb* of any set of elements. The element *glb* POST(τ) is then shown to be a fixedpoint of τ, and it is clearly below all the other fixedpoints of τ (which are all contained in POST(τ)).

(*ii*) (A constructive proof, due to Hitchcock and Park [1972], Cadiou [1972]): This is a rather complicated proof, which constructs a transfinite ascending chain of approximations $\tau^{(\lambda)}[\Omega]$. This chain is shown (by transfinite induction) to converge to the least fixedpoint of τ.

The first approach cannot be directly applied when a model of complete semilattices is considered. If the function *glb* POST(τ) exists, it is the least fixedpoint of τ in this case as well. However, this function need not exist if POST(τ) is empty, since the *glb* operation is defined only over the nonempty subsets of the complete semilattice. We thus have to show that POST(τ) is not empty as a first stage in a Tarski-like proof. Fortunately, the existence theorem of maximal fixedpoints (Theorem 2) implies that FXP(τ) (and thus also POST(τ)) is not empty. We thus get the following indirect proof, in which maximal fixedpoints are used in order to show the existence of a least fixedpoint.

Theorem 3 (*The Least Fixedpoint Theorem*): If τ is a monotonic functional (over a complete semilattice) then FXP(τ) contains a least element.

Proof: By the corollary of Theorem 2, POST(τ) is not empty, and thus $f \equiv glb$ POST(τ) exists. By Lemma 2(*iii*), it is a postfixedpoint of τ, and thus $\tau[f] \sqsubseteq f$. The function $\tau[f]$ is also a postfixedpoint of τ, and thus $f \equiv glb$ POST(τ) $\sqsubseteq \tau[f]$ as well. Consequently $f \equiv \tau[f]$ and therefore $f \in$ FXP(τ). It is the least fixedpoint of τ since $f \equiv glb$ POST(τ) $\sqsubseteq glb$ FXP(τ). Q.E.D.

Theorem 3 can be used in order to find the relationships between prefixedpoints, postfixedpoints and fixedpoints in general. The relative form of Theorem 3 is:

Theorem 4: For a monotonic functional (over a complete semilattice):

(*i*) If f is a prefixedpoint of τ, then there exists a least fixedpoint in the set of functions $S_f = \{g \mid f \sqsubseteq g\}$.

(*ii*) If f is a postfixedpoint of τ, then there exists a greatest fixedpoint in the set of functions $S^f = \{g \mid g \sqsubseteq f\}$.

Proof:

(*i*) Since $f \in \mathrm{PRE}(\tau)$, Theorem 2 guarantees that S_f contains at least one fixedpoint. The proof of Theorem 3 can then be applied without change (over the complete semilattice S_f).

(*ii*) Using the inverse relation, $h_1 \lessdot h_2$ if $h_2 \sqsubseteq h_1$, it can be shown that $(S^f, \lessdot)$ is a complete lattice. Theorem 3 now shows that S^f contains a least fixedpoint with respect to $\lessdot$; this fixedpoint is clearly greatest with respect to $\sqsubseteq$. Q.E.D.

Part II: The Convergence of Functions to Fixedpoints

In Part I we defined our model of recursive definitions and studied its basic properties. Using these results, we now analyze the methods by which fixedpoints of recursive definitions can be "accessed" from other partial functions. In essence, each "access method" uses a given initial function f_0 as a starting point, and constructs a sequence of functions which converges to a fixedpoint of τ. We want the fixedpoint obtained to be "closest" to the initial function. Since the ordering $\sqsubseteq$ is only partial, one can directly compare in this sense only fixedpoints related by $\sqsubseteq$. The most natural definition of this notion is therefore:

Definition: A fixedpoint g of τ is said to be *close* to a partial function f_0 if for every fixedpoint h of τ:

(*i*) if $h \sqsubseteq f_0$ then $h \sqsubseteq g$, and

(*ii*) if $f_0 \sqsubseteq h$ then $g \sqsubseteq h$.

In other words, the fixedpoint g is close to f_0 if it is above any fixedpoint below f_0, and below any fixedpoint above f_0. A priori, it is not clear that such a close fixedpoint must exist for any partial function f_0 -- this will be one of the results proved in this part.

All the functionals considered in this part are term functionals.

4. The Direct Access Method

Kleene's version of the Least Fixedpoint Theorem for continuous functionals shows that by *repeated* application of the functional τ to the initial function Ω, one can construct a sequence $\{\tau^{(i)}[\Omega]\}$ whose limit is the least fixedpoint of τ. This method (which we call the *direct access method*) can be applied to an arbitrary initial function f_0, but in general the sequence obtained need not converge to a limit. The following example demonstrates such a case:

Example 6: Consider the recursive definition over the natural numbers:

$$F(x) \equiv \textit{if } x \geq 10 \textit{ then } F(x-10) \textit{ else } F(x+1)$$

The collection of equalities implied by this recursive definition has a cyclic component:

$$F(0) \equiv F(1) \equiv F(2) \equiv \ldots \equiv F(9) \equiv F(10) \equiv F(0)$$

and the additional equalities:

$$F(11) \equiv F(1), \quad F(12) \equiv F(2), \ldots .$$

It is clear that any constant function is a fixedpoint of the recursive definition and there are no other fixedpoints; the least fixedpoint is Ω, and any constant total function is a maximal fixedpoint.

Consider now the two initial functions:

$$f_1(x) \equiv \begin{cases} 0 & \text{if } x \equiv 0 \\ \omega & \text{otherwise} \end{cases} \qquad f_2(x) \equiv \begin{cases} 0 & \text{if } 0 \leq x \leq 10, \\ 1 & \text{otherwise} \end{cases}$$

The sequence $\{\tau^{(i)}[f_1]\}$ does not converge, since the value 0 is rotated in the cycle $x=0,1,\ldots,10$ under the repeated application of τ. On the other hand, the sequence $\{\tau^{(i)}[f_2]\}$ converges to the fixedpoint *zero* of τ, since all the nonzero values of f_2 are eventually replaced by 0. Note that this sequence is neither an ascending chain nor a descending chain (in fact, no two distinct elements are ever consistent), but it converges according to the generalized notion of *lim*. □

Definition: The function f_0 *converges to* g (under a functional τ) if $\{\tau^{(i)}[f_0]\}$ is a convergent sequence and g is its limit.

We now state and prove the basic result:

Theorem 5: If f_0 converges to g under τ, then g is a fixedpoint which is close to f_0.

Proof: To show that g is a fixedpoint of τ, we use the (generalized) continuity of τ:

$$\tau[g] \equiv \tau[lim\{\tau^{(i)}[f_0]\}] \equiv lim\{\tau[\tau^{(i)}[f_0]]\} \equiv lim\{\tau^{(i+1)}[f_0]\} \equiv g.$$

To show that g is close to f_0, consider an arbitrary fixedpoint h of τ:

(i) If $h \sqsubseteq f_0$ then by the monotonicity of τ, $\tau^{(i)}[h] \sqsubseteq \tau^{(i)}[f_0]$ for all i, and thus since h is a fixedpoint

$$h \equiv lim\{\tau^{(i)}[h]\} \sqsubseteq lim\{\tau^{(i)}[f_0]\} \equiv g .$$

(ii) If $f_0 \sqsubseteq h$ then similarly:

$$g \equiv lim\{\tau^{(i)}[f_0]\} \sqsubseteq lim\{\tau^{(i)}[h]\} \equiv h .$$ Q.E.D.

We can describe the result of Theorem 5 as follows: if g_1 and g_2 are any two fixedpoints of τ such that $g_1 \sqsubseteq f_0 \sqsubseteq g_2$, and if $\{\tau^{(i)}[f_0]\}$ converges, then it converges to a fixedpoint g which is also in the "box" $g_1 \sqsubseteq g \sqsubseteq g_2$. Note that, unless $f_0 \in$ PRE(τ) $\cup$ POST(τ), an initial function f_0 need not be related by $\sqsubseteq$ to the fixedpoint g to which it leads. Furthermore, there need not be a greatest element among the fixedpoints which are less defined than f_0 or a least element among the fixedpoints which are more defined than f_0.

Given an arbitrary initial function f_0, it may be hard to determine in advance whether the sequence $\{\tau^{(i)}[f_0]\}$ converges or not. One important case in which the convergence is guaranteed is when f_0 is either a prefixedpoint or a postfixedpoint of τ. In these cases the generated sequence is a chain, and thus has a *lim*.

We now proceed to characterize two other cases in which the sequence must converge.

Lemma 4: If $f_1 \sqsubseteq f_0 \sqsubseteq f_2$ where f_1 and f_2 both converge to the fixedpoint g of τ, then f_0 also converges to g.

Proof: By the monotonicity of τ, $\tau^{(i)}[f_1] \sqsubseteq \tau^{(i)}[f_0] \sqsubseteq \tau^{(i)}[f_2]$ for any i. The definition of convergence implies that for each $\bar{x}$ there is a natural number j_0 such that

$$\tau^{(j)}[f_1](\bar{x}) \equiv \tau^{(j)}[f_2](\bar{x}) \equiv g(\bar{x}) \quad \text{for all } j \geq j_0 ,$$

and therefore

$$\tau^{(j)}[f_0](\bar{x}) \equiv g(\bar{x}) \quad \text{for all } j \geq j_0 .$$

In other words, the sequence $\{\tau^{(i)}[f_0]\}$ converges to g. Q.E.D.

One immediate corollary of this "sandwich" property is:

Corollary: If $f_0 \sqsubseteq lfxp(\tau)$, then $lim\{\tau^{(i)}[f_0]\} \equiv lfxp(\tau)$.

The least fixedpoint of τ thus has the interesting property that any initial function $f_0 \sqsubseteq lfxp(\tau)$ converges to it under the repeated application of τ (but not necessarily in the form of an ascending chain). Consequently, in order to access other fixedpoints of τ, one must start with initial functions which are already sufficiently defined.

A slightly different type of result is:

Lemma 5: If $f_1 \sqsubseteq f_2$ and $g \equiv lim\{\tau^{(i)}[f_1]\}$ is a total fixedpoint of τ, then f_2 also converges to g.

Proof: By the monotonicity of τ, $\tau^{(i)}[f_1] \sqsubseteq \tau^{(i)}[f_2]$ for all i. Since the sequence $\{\tau^{(i)}[f_1]\}$ converges to g, for any $\bar{x}$ there is a j_0 such that:

$$\tau^{(j)}[f_1](\bar{x}) \equiv g(\bar{x}) \quad \text{for all } j \geq j_0 ,$$

or, in other words:

$$g(\bar{x}) \sqsubseteq \tau^{(j)}[f_2](\bar{x}) \quad \text{for all } j \geq j_0 .$$

Since g is a total function, we obtain:

$$g(\bar{x}) \equiv \tau^{(j)}[f_2](\bar{x}) \quad \text{for all } j \geq j_0 ,$$

and thus $lim\{\tau^{(i)}[f_2]\} \equiv g$. Q.E.D.

Note that the requirement that g is total is essential; it may well happen that a function f_1 converges to a nontotal maximal fixedpoint g, while a function f_2, which is more defined than f_1, does not converge at all.

Taking $f_1 \equiv \Omega$, we obtain an important special case of Lemma 5:

Corollary: If $lfxp(\tau)$ is a total function, then any initial function f_0 converges to $lfxp(\tau)$.

If a recursive definition has only one fixedpoint, then it is clear that the *lim* of any convergent sequence $\{\tau^{(i)}[f_0]\}$ is $lfxp(\tau)$. However, if the unique fixedpoint $lfxp(\tau)$ is not total, there may be initial functions f_0 for which the sequence $\{\tau^{(i)}[f_0]\}$ does not converge at all.

5. General Access Methods

In the previous section we have considered one of the simplest ways by which we can access the fixedpoints of τ -- the repeated application of τ to an initial function f_0. This method may fail to converge when applied to certain initial functions f_0. In this section we investigate some more general access methods, which are later used in order to access fixedpoints of τ from arbitrary initial functions.

5.1 Access Methods

In order to formally introduce the general notion of an access method, we first define:

Definition: The set of *formulae* is defined inductively as follows:

(*i*) The symbol F is a formula (F is said to be a *function variable*).

(*ii*) If $\mathfrak{F}$ is a formula, then $\boldsymbol{\tau}[\mathfrak{F}]$ is a formula ($\boldsymbol{\tau}$ is said to be a *functional variable*).

(*iii*) If $\mathfrak{F}_1$, $\mathfrak{F}_2$ are formulae, then $glb\{\mathfrak{F}_1,\mathfrak{F}_2\}$ and $lub\{\mathfrak{F}_1,\mathfrak{F}_2\}$ are formulae.

Given a formula $\mathfrak{F}$ and a functional τ, we denote by $\mathfrak{F}^\tau$ the formula in which the functional variable $\boldsymbol{\tau}$ is interpreted as τ. $\mathfrak{F}^\tau$ can be considered as a functional (over the same domain of functions as τ) in the following way: Given any function f, $\mathfrak{F}^\tau[f]$ is the function obtained by evaluating the formula $\mathfrak{F}$ in which $\boldsymbol{\tau}$ is interpreted as τ and F is interpreted as f. Unlike the functionals considered so far, $\mathfrak{F}^\tau$ may *fail* when applied to certain functions f, in case the *lub* of inconsistent functions is to be taken during the evaluation process; in this case, $\mathfrak{F}^\tau[f]$ is not defined.

Example 7: Consider the formula:

$$glb\{\boldsymbol{\tau}[lub\{\mathrm{F},\boldsymbol{\tau}[\mathrm{F}]\}],\mathrm{F}\}\ ,$$

and the functional

$$\tau[\mathrm{F}](x):\ \mathrm{F}(x+1)$$

over the natural numbers.

The functional $\mathfrak{F}^{\tau}$ fails for the identity function $f(x) \equiv x$, since f and $\tau[f]$ are inconsistent, and thus their *lub* is not defined. However, $\mathfrak{F}^{\tau}$ does not fail for the function:

$$f(x) \equiv \begin{cases} 0 & \text{if } x \equiv 0 \ (mod\ 3) \\ \omega & \text{otherwise} \end{cases}$$

and the function $\mathfrak{F}^{\tau}[f]$ is Ω. □

Given a functional τ and initial function f, we may consider a function $\mathfrak{F}^{\tau}[f]$ as a modification of f. A sequence of formulae $\{\mathfrak{F}_i\}$ can thus be used in order to construct a sequence of successively modified functions $\{\mathfrak{F}_i^{\tau}[f]\}$. If the sequence $\{\mathfrak{F}_i\}$ is properly chosen, this sequence of functions may converge to a fixedpoint of τ. We thus define:

Definition: An *access method* $\mathfrak{A}$ is a sequence of formulae $\{\mathfrak{F}_i\}$. For a given functional τ, a partial function f is said to *converge to g under* $\mathfrak{A}$ if all the functions $\mathfrak{F}_i^{\tau}[f]$ exist, and $lim\{\mathfrak{F}_i^{\tau}[f]\} \equiv g$. If some of the functions $\mathfrak{F}_i^{\tau}[f]$ do not exist, the method is said to *fail* for τ and f.

In the case the formulae $\mathfrak{F}_i$ become successively more complicated, it is convenient to use a slightly modified notation for formulae. We use a sequence of function variables $F_0, F_1, \ldots$, where each F_i represents the function $\mathfrak{F}_i^{\tau}[f]$, given τ and f. Each function variable F_i is defined by a formula in which all the function variables $F_0, F_1, \ldots, F_{i-1}$, in addition to F, may appear. This representation is equivalent to the original one, since one can always expand the formulae in the new representation to formulae in which only the function variable F may appear.

Some of the simplest access methods, in the new representation, are:

(A) $F_0 \equiv F$

$F_i \equiv \tau[F_{i-1}]$ for $i \geq 1$.

(B) $F_0 \equiv F$

$F_i \equiv glb\{F_{i-1}, \tau^{(i)}[F]\}$ for $i \geq 1$.

(C) $F_0 \equiv F$

$F_i \equiv glb\{F_{i-1}, \tau[F_{i-1}]\}$ for $i \geq 1$.

(D) $F_0 \equiv F$

$F_i \equiv glb\{F, \tau[F_{i-1}]\}$ for $i \geq 1$.

(E) $F_0 \equiv F$

$F_i \equiv \tau[glb\{F, F_{i-1}\}]$ for $i \geq 1$.

Note that methods C-E represent all the nontrivial ways by which F_i can be defined in terms of F_{i-1} and F, using one occurrence of τ and one occurrence of *glb*. Four other simple access methods (denoted by B'-E') can be obtained from methods B-E by replacing each *glb* by *lub*.

Method A is the direct access method discussed in Section 4, since the expanded form of any F_i is $\tau^{(i)}[F]$. Method B is closely related to this method, since each F_i is simply the *glb* of a finite number of powers:

$$F_i \equiv glb\{F, \tau[F], \tau^{(2)}[F], \cdots, \tau^{(i)}[F]\}.$$

For any functional τ and initial function f, the sequence of functions $\{f_i\}$ generated by method B is a descending chain, since the *glb* in the formula for F_{i+1} contains one more term than the *glb* in the formula for F_i. The convergence of any initial function f is thus guaranteed, but unlike the case of the direct access method, the limit function need not be a fixedpoint of τ. This is demonstrated in the following example:

Example 8: Let τ be the following functional over the natural numbers:

$$\tau[F](x): \ \textit{if } x = 0 \textit{ then } F(x)+1 \textit{ else } 0 \cdot F(x-1).$$

Let f be the initial function:

$$f(x) \equiv \begin{cases} 0 & \text{if } x \equiv 0,1 \\ \omega & \text{otherwise}. \end{cases}$$

For any $i \geq 0$,

$$\tau^{(i)}[f](x) \equiv \begin{cases} i & \text{if } x \equiv 0 \\ 0 & \text{if } 1 \leq x \leq i+1 \\ \omega & \text{otherwise}, \end{cases}$$

and thus the *glb* of all these functions is:

$$glb\{\tau^{(i)}[f]\}(x) \equiv \begin{cases} 0 & \text{if } x \equiv 1 \\ \omega & \text{otherwise}. \end{cases}$$

This function is not a fixedpoint of τ (as a matter of fact, it is not even a prefixedpoint or a postfixedpoint of τ). □

5.2 The Descending Access Method

Among the access methods listed above, we shall be interested mainly in method C, called *the descending access method*, and in method C′, called *the ascending access method*. In this section we study the behavior of the first method.

For any initial function f, the descending access method constructs a descending chain of functions $\{f_i\}$, since each f_i is the *glb* of f_{i-1} with some other function. The idea behind the method is to "smooth up" the initial function f by repeatedly taking the common part f_i of the functions f_{i-1} and $\tau[f_{i-1}]$; hopefully such a process may result in a function whose values are preserved under the application of τ, i.e. a fixedpoint of τ.

If the initial function f is a prefixedpoint or a postfixedpoint of τ, then the sequence $\{f_i\}$ generated by method C has an especially simple form:

Lemma 6: Let $\{f_i\}$ be the sequence generated by the descending access method C for τ and f. Then:

(*i*) If $f \in \mathrm{PRE}(\tau)$ then for all i, $f_i \equiv f$.

(*ii*) If $f \in \mathrm{POST}(\tau)$ then for all i, $f_i \equiv \tau^{(i)}[f]$.

Proof:

(*i*) The proof is by induction on i. For $i = 0$, $f_0 \equiv f$ by definition. Suppose that for some i, $f_i \equiv f$. Then:

$$f_{i+1} \equiv glb\{f_i, \tau[f_i]\} \equiv glb\{f, \tau[f]\} \equiv f,$$

since $f \sqsubseteq \tau[f]$.

(*ii*) This part is again proved by induction. For $i = 0$, $f_0 \equiv f$ by definition. If for some i, $f_i \equiv \tau^{(i)}[f]$, then f_i is also a postfixedpoint of τ by Lemma 2(*i*), and thus:

$$f_{i+1} \equiv glb\{f_i, \tau[f_i]\} \equiv \tau[f_i] \equiv \tau[\tau^{(i)}[f]] \equiv \tau^{(i+1)}[f].$$ Q.E.D.

Part (*i*) of Lemma 6 shows that an initial function f may converge under method C to a limit function which is not a fixedpoint of τ. However, we have:

Theorem 6: For any functional τ and initial function f, the sequence $\{f_i\}$ generated by the descending access method C converges to a prefixedpoint of τ. This limit function is the greatest among the prefixedpoints of τ that are below f.

Proof: The fact that the descending chain $\{f_i\}$ converges to some limit function g, which is below f, is clear. We now show that g is a prefixedpoint of τ, i.e. $g \sqsubseteq \tau[g]$. By definition

$$g \equiv lim\{f_i\} \equiv lim\{glb\{f_{i-1}, \tau[f_{i-1}]\}\}.$$

Since both $\{f_{i-1}\}$ and $\{\tau[f_{i-1}]\}$ are convergent sequences

$$g \equiv glb\{lim\{f_{i-1}\}, lim\{\tau[f_{i-1}]\}\},$$

and by the continuity of τ and the definition of g

$$g \equiv glb\{lim\{f_{i-1}\}, \tau[lim\{f_{i-1}\}]\} \equiv glb\{g, \tau[g]\}.$$

The fact that $g \sqsubseteq \tau[g]$ follows now from the equality $g \equiv glb\{g, \tau[g]\}$.

Finally, we show that if h is any prefixedpoint of τ such that $h \sqsubseteq f$, then $h \sqsubseteq g$. It suffices to show that $h \sqsubseteq f_i$ for all i. We prove this by induction on i.

If $i = 0$, then $f_0 \equiv f$ and thus $h \sqsubseteq f_0$ by assumption. If f_i satisfies $h \sqsubseteq f_i$ for some i, then:

$$h \sqsubseteq \tau[h] \sqsubseteq \tau[f_i],$$

and thus h is below both f_i and $\tau[f_i]$, implying that

$$h \sqsubseteq glb\{f_i, \tau[f_i]\} \equiv f_{i+1}.$$

Q.E.D.

The existence of a greatest prefixedpoint below an arbitrary partial function f can be independently proved by taking the *lub* of the consistent set of all the prefixedpoints of τ below f, and using the fact that this *lub* is itself a prefixedpoint of τ. Theorem 6 shows that the descending access method always leads to this greatest prefixedpoint. Note that the set of *fixedpoints* below f need not have a greatest element (in fact, it may even be empty if $f \equiv lfxp(\tau)$).

We can now show that the descending access method is the least access method in the following sense:

Theorem 7: **For any functional τ, if an initial function f converges to g_1 under the descending access method C and to g_2 under some other access method $\mathfrak{A}$, then $g_1 \sqsubseteq g_2$.**

Proof: **We first prove that for any formula $\mathfrak{F}$ for which $\mathfrak{F}^\tau[f]$ exists, $g_1 \sqsubseteq \mathfrak{F}^\tau[f]$. The proof is by induction on the structure of the formula $\mathfrak{F}$.**

(*i*) If $\mathfrak{F}$ is F, then clearly $g_1 \sqsubseteq f \equiv \mathfrak{F}^\tau[f]$.

(*ii*) If $\mathfrak{F}$ is of the form $\tau[\mathfrak{F}_1]$, then by the induction hypothesis $g_1 \sqsubseteq \mathfrak{F}_1{}^\tau[f]$. Since by Theorem 6, g_1 is a prefixedpoint of τ, we have:

$$g_1 \sqsubseteq \tau[g_1] \sqsubseteq \tau[\mathfrak{F}_1^\tau[f]] \equiv \mathfrak{F}^\tau[f].$$

(iii) if $\mathfrak{F}$ is of the form $glb\{\mathfrak{F}_1,\mathfrak{F}_2\}$ then $g_1 \sqsubseteq \mathfrak{F}_1^{\tau}[f]$ and $g_1 \sqsubseteq \mathfrak{F}_2^{\tau}[f]$ by the induction hypothesis, and thus

$$g_1 \sqsubseteq glb\{\mathfrak{F}_1^{\tau}[f],\mathfrak{F}_2^{\tau}[f]\} \equiv \mathfrak{F}^{\tau}[f].$$

(iv) If $\mathfrak{F}$ is of the form $lub\{\mathfrak{F}_1,\mathfrak{F}_2\}$ then

$$g_1 \sqsubseteq \mathfrak{F}_1^{\tau}[f] \sqsubseteq lub\{\mathfrak{F}_1^{\tau}[f],\mathfrak{F}_2^{\tau}[f]\} \equiv \mathfrak{F}^{\tau}[f].$$

The *lub* exists since we assume that $\mathfrak{F}^{\tau}[f]$ is defined.

Let $\mathfrak{A}$ be the sequence of formulae $\{\mathfrak{F}_i\}$. The functions $\mathfrak{F}_i^{\tau}[f]$ exist since we assume that this sequence converges to g_2. Since $g_1 \sqsubseteq \mathfrak{F}_i^{\tau}[f]$ for all i, and the sequence $\{\mathfrak{F}_i^{\tau}[f]\}$ is convergent,

$$g_1 \sqsubseteq lim\{\mathfrak{F}_i^{\tau}[f]\} \equiv g_2.$$ Q.E.D.

Using Theorems 6 and 7, we can now indirectly show that access methods C and D are equivalent. One can easily show that any initial function f converges under method D to some prefixedpoint g_1 of τ. If we denote by g_2 the prefixedpoint to which f converges under the descending access method C, then $g_2 \sqsubseteq g_1$ by Theorem 6, and $g_1 \sqsubseteq g_2$ by Theorem 7. Consequently, any initial function f converges to the same function under access methods C and D.

5.3 The Ascending Access Method

In this section we consider the ascending access method C′, which is dual to the descending access method C. The following results (which are stated without proofs) are analogous to those obtained in subsection 5.2; the main difference is that access methods in which the *lub* operation occurs may fail if the *lub* of inconsistent functions is taken.

Lemma 7: Let $\{f_i\}$ be a sequence of functions generated by the ascending access method C′ for τ and f. Then:

(i) If $f \in \mathrm{PRE}(\tau)$ then for all i, $f_i \equiv \tau^{(i)}[f]$

(*ii*) If $f \in \mathrm{POST}(\tau)$ then for all i, $f_i \equiv f$.

Theorem 8: For any functional τ and initial function f, if the functions f_i generated by the ascending access method C′ exist, then the sequence $\{f_i\}$ converges to a postfixedpoint of τ. This limit function is the least among the postfixedpoints of τ that are above f.

Theorem 9: For any functional τ, if an initial function f converges to g_1 under the ascending access method C′ and to g_2 under some other access method $\mathfrak{A}$, then $g_2 \sqsubseteq g_1$.

The following Lemma gives a sufficient condition on τ and f which guarantees the existence of $\mathfrak{F}^{\tau}[f]$ for an arbitrary formula $\mathfrak{F}$.

Lemma 8: For a given τ and f, if there is a postfixedpoint g of τ which satisfies $f \sqsubseteq g$, then for any formula $\mathfrak{F}$, the function $\mathfrak{F}^{\tau}[f]$ exists.

Proof: We show (by induction on the structure of $\mathfrak{F}$) that $\mathfrak{F}^{\tau}[f]$ exists and satisfies $\mathfrak{F}^{\tau}[f] \sqsubseteq g$ for any formula $\mathfrak{F}$:

(*i*) If $\mathfrak{F}$ is of the form F, then $\mathfrak{F}^{\tau}[f] \equiv f \sqsubseteq g$ by assumption.

(*ii*) If $\mathfrak{F}$ is of the form $\tau[\mathfrak{F}_1]$, then by the induction hypothesis $\mathfrak{F}_1^{\tau}[f]$ exists and satisfies $\mathfrak{F}_1^{\tau}[f] \sqsubseteq g$, and thus:

$$\mathfrak{F}^{\tau}[f] \equiv \tau[\mathfrak{F}_1^{\tau}[f]] \sqsubseteq \tau[g] \sqsubseteq g.$$

(*iii*) If $\mathfrak{F}$ is of the form $glb\{\mathfrak{F}_1,\mathfrak{F}_2\}$, where $\mathfrak{F}_1^{\tau}[f] \sqsubseteq g$ and $\mathfrak{F}_2^{\tau}[f] \sqsubseteq g$, then clearly:

$$\mathfrak{F}^{\tau}[f] \equiv glb\{\mathfrak{F}_1^{\tau}[f],\mathfrak{F}_2^{\tau}[f]\} \sqsubseteq glb\{g, g\} \equiv g.$$

(*iv*) Similarly, if $\mathfrak{F}$ is of the form $lub\{\mathfrak{F}_1,\mathfrak{F}_2\}$, where $\mathfrak{F}_1^{\tau}[f] \sqsubseteq g$ and $\mathfrak{F}_2^{\tau}[f] \sqsubseteq g$, then these two functions are consistent, and thus their *lub* exists and satisfies:

$$\mathfrak{F}^{\tau}[f] \equiv lub\{\mathfrak{F}_1^{\tau}[f], \mathfrak{F}_2^{\tau}[f]\} \sqsubseteq lub\{g, g\} \equiv g.$$

Q.E.D.

Corollary: For a given τ and f, if there is a postfixedpoint g of τ which satisfies $f \sqsubseteq g$, then no access method $\mathfrak{A}$ can fail for τ and f.

Note that this corollary does not imply that such an f converges to a limit under $\mathfrak{A}$.

The sufficient condition in this corollary is clearly not necessary in general. Consider, for example, the following access method:

$$F_0 \equiv glb\{F, \tau[F]\}$$
$$F_1 \equiv glb\{\tau[F], \tau^2[F]\}$$
$$F_i \equiv \tau[lub\{F_{i-1}, \tau[F_{i-2}]\}] \quad \text{for } i \geq 2.$$

For any functional τ and initial function f, all the pairs of functions f_{i-1}, $\tau[f_{i-2}]$ to which the *lub* is applied are consistent, and thus this access method can never fail.

We now show that for the special case of the ascending access method, the condition in Lemma 8 exactly characterizes the cases in which the method does not fail.

Lemma 9: A necessary and sufficient condition for a function f to converge under the ascending access method C′ is the existence of a postfixedpoint g of τ such that $f \sqsubseteq g$.

Proof: If the postfixedpoint g exists, then by the corollary of Lemma 8 the sequence $\{f_i\}$ is defined. Since it is an ascending chain, it is a convergent sequence and thus f converges under method C′.

On the other hand, if f converges under C′ then, by Theorem 8, the limit g of the generated sequence $\{f_i\}$ is a postfixedpoint of τ. Furthermore, $f \sqsubseteq g$, since $\{f_i\}$ is an ascending chain whose first element is f. We have thus shown the existence of a postfixedpoint g of τ which satisfies $f \sqsubseteq g$.

Q.E.D.

By the corollary of Lemma 8 and by Lemma 9, the ascending access method C′ is the most exacting in the sense that:

Corollary: If method C′ does not fail for a given τ and f, then no other access method $\mathfrak{A}$ can fail for τ and f.

6. The Fixedpoint Method

In this section we finally devise a method which always succeeds and under which any initial function converges to a fixedpoint. As we show in subsection 6.2, no single access method can achieve this goal; we thus need a somewhat more complicated method, based on compositions of access methods. This notion is formally defined as follows.

Definition: For a functional τ, an initial function f is said to *converge to* h *under the composition* $\mathfrak{A}_2 \circ \mathfrak{A}_1$ of two access methods $\mathfrak{A}_1$ and $\mathfrak{A}_2$, if f converges to some function g under $\mathfrak{A}_1$ and g converges to h under $\mathfrak{A}_2$.

This definition can be naturally extended to an n-fold composition $\mathfrak{A}_n \circ \ldots \circ \mathfrak{A}_2 \circ \mathfrak{A}_1$.

6.1 Properties of the Fixedpoint Method

Definition: The *fixedpoint method* is the composition $A \circ C$ of the two access methods C and A.

The main result concerning the fixedpoint method is:

Theorem 10: For a functional τ, any initial function f converges under the fixedpoint method $A \circ C$ to a fixedpoint of τ which is close to f. Furthermore, this fixedpoint is the least among all the fixedpoints of τ which can be reached from f under any composition of access methods.

Proof: Any initial function f converges under $A \circ C$ to a fixedpoint h of τ, since f converges under C to a prefixedpoint g of τ (by Theorem 6), and g converges under A to a fixedpoint h of τ (by Theorem 5).

We now show that h is close to the initial function f. Let l be an arbitrary fixedpoint of τ. Then:

(*i*) If $l \sqsubseteq f$, the prefixedpoint l is below f, and by Theorem 6, the prefixedpoint g to which f converges under C satisfies $l \sqsubseteq g$. Consequently,

$$l \equiv lim\{\tau^{(i)}[l]\} \sqsubseteq lim\{\tau^{(i)}[g]\} \equiv h$$

and thus $l \sqsubseteq h$.

(*ii*) If $f \sqsubseteq l$, then clearly $g \sqsubseteq l$, since $g \sqsubseteq f$. This implies that:

$$h \equiv lim\{\tau^{(i)}[g]\} \sqsubseteq lim\{\tau^{(i)}[l]\} \equiv l,$$

and thus $h \sqsubseteq l$.

Finally, we show that h is the least among all the fixedpoints of τ which can be reached from f under any composition of access methods.

Suppose that f converges to a fixedpoint l of τ under the composition $\mathfrak{A}_n \circ \mathfrak{A}_{n-1} \circ \ldots \circ \mathfrak{A}_1$ of access methods. Let us denote by g_i $(i=1,...,n)$ the successive limit functions to which f converges under the partial compositions $\mathfrak{A}_i \circ \ldots \circ \mathfrak{A}_1$ (in particular, $g_n \equiv l$). The function f converges to the prefixedpoint g under C. We now show that $g \sqsubseteq g_i$ for all $i=1,...,n$.

Since f converges to g and g_1 under the respective methods C and $\mathfrak{A}_1$, we have (by Theorem 7) that $g \sqsubseteq g_1$. The function g_1 converges to g_2 under $\mathfrak{A}_2$, and to some prefixedpoint g_2' under C (this convergence is assured since *any* initial function converges under C). By Theorem 6, g_2' is the greatest among the prefixedpoints of τ which are below g_1. However, g is one such prefixedpoint and thus $g \sqsubseteq g_2'$. On the other hand, $g_2' \sqsubseteq g_2$ by Theorem 7; we thus conclude that $g \sqsubseteq g_2$.

Continuing this type of reasoning for $i=3,...,n$, we can show that $g \sqsubseteq g_i$ for all i. In particular, g_n is the fixedpoint l of τ, and thus $g \sqsubseteq l$.

We still have to show the relation $h \sqsubseteq l$ between the fixedpoints h and l obtained under the compositions A $\circ$ C and $\mathfrak{A}_n \circ \ldots \circ \mathfrak{A}_1$, respectively. We already know that $g \sqsubseteq l$, and that the prefixedpoint g converges to h under the direct access method A. By Theorem 5, the fixedpoint h is close to g, and in particular $h \sqsubseteq k$ for any fixedpoint k of τ satisfying $g \sqsubseteq k$. Since l is one such fixedpoint, we obtain the desired result $h \sqsubseteq l$.

Q.E.D.

An initial function f which converges under the ascending access method C′, converges to a

postfixedpoint g of τ (by Theorem 8). The function g is assured to converge to a fixedpoint h of τ under the direct access method A, and thus any f converges under A $\circ$ C′ to a fixedpoint of τ, provided only that method C′ does not fail for f. By Lemma 8, this condition is equivalent to the existence of a postfixedpoint of τ above f. The dual to Theorem 10 is therefore:

Theorem 11: For any functional τ and initial function f such that there exists a postfixedpoint of τ above f, the function f converges under A $\circ$ C′ to a fixedpoint of τ which is close to f. Furthermore, this fixedpoint is the greatest among all the fixedpoints of τ which can be reached from f under any composition of access methods.

The proof of Theorem 11 is analogous to the proof of Theorem 10; the additional assumption about the existence of a postfixedpoint is used only in order to establish the existence of the appropriate limits.

Two other compositions of access methods which are equivalent to A $\circ$ C and A $\circ$ C′ are characterized in the following lemma:

Lemma 10:

(*i*) For any τ and f, f converges to the same function under A $\circ$ C and C′ $\circ$ C.

(*ii*) For any τ and f, f converges to the same function under A $\circ$ C′ and C $\circ$ C′, provided that C′ does not fail.

Proof:

(*i*) The function g to which f converges under C is a prefixedpoint of τ. By Lemma 7(*i*), methods A and C′ behave in the same way for prefixedpoints, and thus the compositions A $\circ$ C and C′ $\circ$ C are equivalent.

(*ii*) Similar, by Lemma 7(*ii*). Q.E.D.

An arbitrary initial function f can be considered as a "distorted fixedpoint" to which two types of corrections must be applied:

(*i*) Defined parts, which are either changed or replaced by ω under the application of τ, must be deleted from the function since they do not represent possible fixedpoint values.

(*ii*) Undefined parts, which are replaced by defined values under the application of τ, must be completed with the appropriate fixedpoint values.

The descending access method performs only the first type of correction, while the ascending access method performs only the second type of correction. None of them can transform an arbitrary initial function f to a fixedpoint of τ, but when both of them are applied to f, a fixedpoint of τ is obtained. The order in which the two correcting stages are performed (i.e., $C' \circ C$ or $C \circ C'$) may affect the fixedpoint obtained, since the two access methods C and C′ do not commute in general. Furthermore, the composition $C \circ C'$ in which the deletion stage comes after the completion stage may fail, while the fixedpoint method $C' \circ C$ cannot.

Let us denote by S_f^τ the set of fixedpoints of τ which can be reached from f by compositions of access methods. The following immediate corollaries summarize the structure of S_f^τ in the case where method C′ does not fail for τ and f.

Corollaries:

(*i*) The set S_f^τ contains a least element (accessed by $C' \circ C$) and a greatest element (accessed by $C \circ C'$).

(*ii*) If f converges to the same function h under $C' \circ C$ and $C \circ C'$, then h is the only fixedpoint of τ which can be reached from f (by any composition of access methods).

(*iii*) If f is either a prefixedpoint or a postfixedpoint of τ, which converges to h under the direct access method A, then h is the only fixedpoint of τ which can be reached from f (by any composition of access methods).

(*iv*) If f is a fixedpoint of τ, then f converges to itself under any composition of access methods.

(*v*) All the fixedpoints in S_f^τ are close to f (however there may be other fixedpoints which are close to f but which are inaccessible from f by any composition of access methods).

(*vi*) All the fixedpoints in S_f^τ are consistent with the initial function f.

If access method C' fails for τ and f, then the set S_f^τ need not have a greatest element, and the functions in S_f^τ need not be consistent with f. However, if f is either a prefixedpoint or a postfixedpoint of τ, then C' cannot fail for τ and f.

Theorem 10 guarantees that for any initial function f, there is at least one fixedpoint h of τ which is close to f. For a fixed functional τ, we can consider the fixedpoint method $A \circ C$ as a functional $\mathfrak{M}_\tau$ which maps any function f to some fixedpoint of τ that is close to f. The functional $\mathfrak{M}_\tau$ maps the set PF of partial functions (over the appropriate domain) *onto* the set $FXP(\tau)$, since any fixedpoint h of τ is mapped to itself under $\mathfrak{M}_\tau$. Our aim in the rest of this subsection is to study the monotonicity and continuity properties of $\mathfrak{M}_\tau$.

Theorem 12: For any functional τ, $\mathfrak{M}_\tau : PF \to FXP(\tau)$ is monotonic.

Proof: By induction on the structure of formulae it is easy to show that for a fixed functional τ, any access method is a monotonic mapping from initial functions to limit functions (whenever they exist). Consequently, the composition $A \circ C$ (for which limits always exist) is also monotonic.

Q.E.D.

Note that the *existence* of such a monotonic mapping from PF onto $FXP(\tau)$ is not surprising (due to the many structural similarities between the two sets); however, the theory of access methods enables us to define the mapping in a simple and constructive way.

The functional $\mathfrak{M}_\tau$ whose monotonicity was shown above, is not continuous. This fact does not stem from the special way in which $\mathfrak{M}_\tau$ is defined. The following theorem shows that for certain functionals τ, any such mapping is inherently noncontinuous.

Theorem 13: There are functionals τ, for which any mapping Θ: $PF \to FXP(\tau)$, which assigns to each partial function f a fixedpoint of τ that is close to f, must be noncontinuous.

Proof: Let τ be the following functional over the integers:

$$\tau[F](x) : \textit{if } F(x-1) = 0 \textit{ then } F(x) + 0 \cdot F(x+1) + 0 \cdot x$$
$$\textit{else } F(x-1) + 0 \cdot F(x+1) + 0 \cdot x .$$

The special property of this functional is that for a certain sequence $\{f_i\}$ of initial functions, each f_i has exactly one fixedpoint -- Ω -- which is close to it. By the assumption on Θ,

$\Theta[f_i] \equiv \Omega$ for all i, and thus $lim\{\Theta[f_i]\} \equiv \Omega$. We shall use this fact in order to show that Θ does not preserve the *lim* of convergent sequences.

The two subterms $0 \cdot x$ in the functional guarantee that any fixedpoint of τ is undefined for $x \equiv \omega$. For other values of x, $\tau[F](x)$ is defined in terms of both $F(x-1)$ and $F(x+1)$, and thus any fixedpoint of τ is either Ω or total over the defined integers. Among the total functions, only two types of functions are fixedpoints of τ:

(*i*) The constant functions:

$$g(x) \equiv c \quad \text{for some defined integer } c;$$

(*ii*) The split-constant functions:

$$g(x) \equiv \begin{cases} 0 & \text{if } x \le j \\ c & \text{if } x > j \end{cases} \qquad \text{for some defined integers } c \text{ and } j.$$

Consider now the ascending chain of functions $\{f_i\}$, where

$$f_i(x) \equiv \begin{cases} 0 & \text{if } x \le i \\ \omega & \text{otherwise.} \end{cases}$$

Each f_i is a postfixedpoint of τ, which descends to the fixedpoint Ω of τ under the direct access method A. We now show that Ω is the unique fixedpoint of τ which is close to f_i.

Let h be a fixedpoint of τ which is close to f_i. By definition, h must be below any fixedpoint of τ which is above f_i. Two such fixedpoints above f_i are:

$$g_1(x) \equiv 0$$

$$g_2(x) \equiv \begin{cases} 0 & \text{if } x \le i \\ 1 & \text{if } x > i. \end{cases}$$

The only fixedpoint of τ which is below both g_1 and g_2 is Ω, since no other nontotal function can be a fixedpoint of τ. On the other hand, one can easily show that Ω itself is a fixedpoint which is close to f_i. We have thus shown that Ω is the *unique* fixedpoint of τ which is close to f_i. Using the assumption on Θ, we can now deduce:

$$\Theta[f_i] \equiv \Omega \quad \text{for all } i .$$

Let us consider now the function $zero \equiv lim\{f_i\}$. Since *zero* is a fixedpoint of τ , it is the unique fixedpoint of τ which is close to itself, and thus:

$$\Theta[lim\{f_i\}] \equiv \Theta[zero] \equiv zero .$$

We have thus shown that Θ does not preserve the limit of convergent sequences (or even the *lub* of ascending chains).

Q.E.D.

6.2 The Insufficiency of a Single Access Method

Theorem 10 showed that the composition A $\circ$ C of access methods has the interesting property that any initial function converges to a fixedpoint under it. A natural question is whether there exists some single access method $\mathfrak{A}$ which has this property, i.e., whether the fixedpoints of τ can be reached from arbitrary initial functions by means of a single limiting process.

A plausible candidate for such an access method is:

$$F_0 \equiv F$$

$$\left.\begin{array}{l} F_{2i+1} \equiv \tau[F_{2i}] \\ F_{2i+2} \equiv glb\{F_{2i+1}, \tau[F_{2i+1}]\} \end{array}\right\} \text{ for all } i \geq 0 .$$

In this method, the functions with odd indices are defined as in method A , and the functions with even indices are defined as in method C . Unfortunately, one can easily show that certain initial functions f do not converge under this "alternating access method."

In this section we formally prove that any such attempt to construct a single access method, in which any f converges to a fixedpoint, must fail. It suffices to consider for this purpose the simple functional $\tau_0[F](x) : F(x+1)$ over the natural numbers. What we actually show is that for any "candidate" access method $\mathfrak{A}$, one can construct an appropriate initial function f such that f does not converge to a fixedpoint of τ_0 under $\mathfrak{A}$.

Two useful properties of the selected functional $\tau_0[F](x) : F(x+1)$ are

(*i*) For any two functions f_1, f_2 :

$$\tau_0[glb\{f_1,f_2\}] \equiv glb\{\tau_0[f_1],\tau_0[f_2]\},$$

(*ii*) For any two consistent functions f_1, f_2 :

$$\tau_0[lub\{f_1,f_2\}] \equiv lub\{\tau_0[f_1],\tau_0[f_2]\}.$$

Let $\mathfrak{F}$ be an arbitrary formula. The interpreted formula $\mathfrak{F}^{\tau_0}$ is a composition of τ_0, *glb* and *lub*, and τ_0 commutes with both the *glb* and *lub* operations. We can thus push each occurrence of τ_0 in $\mathfrak{F}_i^{\tau_0}$ all the way inwards, and obtain a modified formula in which various powers of τ_0 are combined by a structure of *glb* and *lub* operations.

Example 9: Consider the formula $\mathfrak{F}$:

$$\tau[lub\{F,\tau[glb\{F,\tau[F]\}]\}].$$

For the special case of the functional τ_0, $\mathfrak{F}^{\tau_0}$ can be transformed in the following way:

$$\tau_0[lub\{F,\tau_0[glb\{F,\tau_0[F]\}]\}] \rightarrow$$
$$\tau_0[lub\{F,glb\{\tau_0[F],\tau_0^{(2)}[F]\}\}] \rightarrow$$
$$lub\{\tau_0[F],\tau_0[glb\{\tau_0[F],\tau_0^{(2)}[F]\}]\} \rightarrow$$
$$lub\{\tau_0[F],glb\{\tau_0^{(2)}[F],\tau_0^{(3)}[F]\}\}.$$

In this modified formula, there are three powers of τ_0 (τ_0, $\tau_0^{(2)}$, $\tau_0^{(3)}$); these powers are connected by a structure consisting of one *glb* and one *lub* operation. □

For a formula $\mathfrak{F}^{\tau_0}$, we define the depth of $\mathfrak{F}^{\tau_0}$, $d(\mathfrak{F}^{\tau_0})$, to be the greatest power of τ_0 occurring in the modified formula. Since $\tau_0^{(k)}[f](x) \equiv f(x+k)$, the value of $\mathfrak{F}^{\tau_0}[f](x)$ is totally determined by the values of $f(x')$ for $x \le x' \le x+d(\mathfrak{F}^{\tau_0})$. We shall later use the fact that any change in the values of $f(x')$ for other arguments x' cannot affect the value of $\mathfrak{F}^{\tau_0}[f](x)$.

We can now prove the theorem:

Theorem 14: Let τ_0 be the following *functional over the natural numbers:*

$$\tau_0[F](x) : F(x+1).$$

Then there is no single access method $\mathfrak{A}$ under which any initial function f converges to a fixedpoint of τ_0.

Proof: We first give an informal overview of the proof. Suppose that the theorem is not true, and access method $\mathfrak{A} = \{\mathfrak{F}_i\}$, has the desired property. We derive a contradiction by constructing an initial function f in such a way that for some ascending sequence $i_0 < i_1 < ...$ of indices,

$$\mathfrak{F}_{i_k}^{\tau}{}^{\circ}[f](0) \equiv \begin{cases} \omega & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd}. \end{cases}$$

The sequence of functions $\{\mathfrak{F}_i^{\tau}{}^{\circ}[f]\}$ thus cannot converge, since it changes value infinitely many times at $x \equiv 0$.

The function f is defined as the *lim* of some convergent sequence of functions $\{g_j\}$. This sequence satisfies, for each k:

$$\mathfrak{F}_{i_k}^{\tau}{}^{\circ}[g_k](0) \equiv \begin{cases} \omega & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd}. \end{cases}$$

For any fixed function g_k, the other functions $g_{k'}$ for $k' > k$, are constructed in such a way that $g_k(x)$ and $g_{k'}(x)$ are identical for all $0 \le x \le d(\mathfrak{F}_{i_k}^{\tau}{}^{\circ})$. Consequently, the limit f of $\{g_j\}$ also satisfies:

$$f(x) \equiv g_k(x) \text{ for all } 0 \le x \le d(\mathfrak{F}_{i_k}^{\tau}{}^{\circ}).$$

Since the value of $\mathfrak{F}_{i_k}^{\tau}{}^{\circ}[g_k](0)$ depends only on the value of g_k for the first $d(\mathfrak{F}_{i_k}^{\tau}{}^{\circ})$ arguments, we obtain:

$$\mathfrak{F}_{i_k}^{\tau\circ}[f](0) \equiv \mathfrak{F}_{i_k}^{\tau\circ}[g_k](0) .$$

This equality establishes the oscillating nature of the sequence of values $\{\mathfrak{F}_{i_k}^{\tau\circ}[f](0)\}$, which is the desired result.

We now formally define the convergent sequence of functions $\{g_j\}$ and the ascending sequence of indices $\{i_j\}$.

As first elements in these sequences, we take $g_0 \equiv \Omega$ and $i_0 \equiv 0$. We justify this selection by noting that $\mathfrak{F}_0^{\tau\circ}[\Omega](0) \equiv \omega$, since Ω is a fixedpoint of τ and thus for any formula $\mathfrak{F}$, $\mathfrak{F}^{\tau\circ}[\Omega] \equiv \Omega$.

We now proceed to define g_1 and i_1. As discussed above, we want $g_1(x)$ to be identical to $g_0(x)$ for any $0 \leq x \leq d(\mathfrak{F}_{i_k}^{\tau\circ})$. We thus define:

$$g_1(x) \equiv \begin{cases} g_0(x) & \text{if } 0 \leq x \leq d(\mathfrak{F}_{i_k}^{\tau\circ}) \\ 0 & \text{otherwise .} \end{cases}$$

By assumption, any initial function converges under $\mathfrak{A}$ to a fixedpoint of τ_0, and thus g_1 converges under $\mathfrak{A}$ to some fixedpoint h of τ. Since g_1 converges to the same fixedpoint *zero* under the two extreme compositions $C' \circ C$ and $C \circ C'$, the function h must be *zero*. By definition of convergence, there is some index i_1 such that

$$\mathfrak{F}_{i_1}^{\tau\circ}[g_1](0) \equiv 0 ,$$

and we have thus found the second function g_1 and second index i_1.

We now briefly outline the next stage in the construction of $\{g_j\}$ and $\{i_j\}$ (i.e., g_2 and i_2). Let m_2 be defined as:

$$m_2 \equiv max(\ 2\ ,\ d(\mathfrak{F}_{i_0}^{\tau\circ})\ ,\ d(\mathfrak{F}_{i_1}^{\tau\circ})\ ) .$$

The function g_2 is defined as:

$$g_2(x) \equiv \begin{cases} g_1(x) & \text{if } 0 \le x \le m_2 \\ \omega & \text{otherwise .} \end{cases}$$

This function converges to Ω under both compositions $C' \circ C$ and $C \circ C'$, and thus g_2 converges to Ω under $\mathfrak{A}$ as well. This convergence implies the existence of an index $i_2 > i_1$ such that

$$\mathfrak{F}_{i_2}^{\tau_0}[g_2] \equiv \omega .$$

The other functions g_k in the sequence are constructed by taking an appropriate initial segment of g_{k-1} and changing the value of the constant tail from 0 to ω or from ω to 0 (according to the oddity of k). The boundary of the initial segment, m_k, is defined in such a way that $m_k \ge k$, and thus the sequence $\{g_j\}$ of functions is assured to converge at any argument x (since $g_k(x)$ is constant for all $k \ge x$). The function $f \equiv lim\{g_j\}$ is thus defined, and by its definition, it satisfies:

$$\mathfrak{F}_{i_k}^{\tau_0}[f](0) \equiv \mathfrak{F}_{i_k}^{\tau_0}[g_k](0) \equiv \begin{cases} \omega & \text{if } x \text{ is even} \\ 0 & \text{if } x \text{ is odd .} \end{cases}$$

Q.E.D.

Future Research

This paper covers only the lattice-theoretical aspects of access methods. Other problems which might be of interest include the computability aspects of access methods, the relations between access methods and substitution/simplification techniques for evaluating fixedpoints, and characterizations of those cases in which a single access method is sufficient in order to access fixedpoints.

Acknowledgment

We would like to thank Nachum Dershowitz and Richard Waldinger for their careful critical reading of the manuscript.

References

Berry, G. [March 1976], *Bottom-up computations of recursive programs*, R.A.I.R.O. Informatique Theorique, Vol. 10, No. 3.

Cadiou, J.M. [March 1972], *Recursive definitions of partial functions and their computations*, Ph.D. Thesis, Computer Science Dept., Stanford University, Stanford, Cal.

Dugundji, J. [1966], *Topology*, Allyn and Bacon, Boston.

Hitchcock, P. and D. Park [July 1972], *Induction rules and proofs of termination*, Proceedings of the IRIA Colloquium on Automata, Languages and Programming Theory, North-Holland.

Kleene, S.C. [1952], *Introduction to Metamathematics*, D. Van Nostrand Inc., Princeton, N.J.

Manna, Z. and A. Shamir [Sept. 1976], *The theoretical aspects of the optimal fixedpoint*, SIAM J. of Computing, Vol. 5, No. 3, pp. 414-426.

Shamir A. [Oct. 1976], *The fixedpoints of recursive definitions*, Ph.D. Thesis, Applied Mathematics Dept., The Weizmann Institute, Rehovot, Israel.

Tarski, A. [1955], *A lattice-theoretical fixpoint theorem and its applications*, Pacific J. of Mathematics, Vol. 5, pp. 285-309.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
14 STAN-CS-77-614, AIM-299

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
6 The Convergence of Functions to Fixedpoints of Recursive Definitions.

5. TYPE OF REPORT & PERIOD COVERED
9 Technical rept.,

6. PERFORMING ORG. REPORT NUMBER
AIM-299

7. AUTHOR(s)
10 Zohar/Manna [redacted] Adi/Shamir

8. CONTRACT OR GRANT NUMBER(s)
15 MDA9Ø3-76-C-Ø2Ø6, ARPA Order-2494

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Artificial Intelligence Laboratory
Stanford University
Stanford, California 94305

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
ARPA Order 2494

11. CONTROLLING OFFICE NAME AND ADDRESS
Eugene Stubbs
ARPA/PM
1400 Wilson Blvd., Arlington, VA 22209

12. REPORT DATE
11 May [redacted]77

13. NUMBER OF PAGES
96 12 49p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
Philip Surra, ONR Representative
Durand Aeronautics Building Room 165
Stanford University
Stanford, California 94305

15. SECURITY CLASS. (of this report)
15

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Releasable without limitations on dissemination.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The classical method for constructing the least fixedpoint of a recursive definition is to generate a sequence of functions whose initial element is the totally undefined function and which converges to the desired least fixedpoint. This method, due to Kleene, cannot be generalized to allow the construction of other fixedpoints.

In this paper we present an alternate definition of convergence and a new fixedpoint access method of generating sequences of functions for a given recursive definition. The initial function of the sequence can be an arbitrary function, and the sequence will always converge to a fixedpoint that is "close" to the initial function. This defines a monotonic mapping from the set of partial functions onto the set of all fixedpoints of the given recursive definition.

Ø94 12Ø Du